SUBDIVIDING A DIGITAL DENTITION MODEL

RELATED APPLICATIONS

This application is a continuation-in-part of U.S. patent application 09/169,276, filed on October 8, 1998, and entitled "Computer Automated Development of an Orthodontic Treatment Plan and Appliance," now abandoned which ~~claims priority from~~ is a continuation-in-part of PCT application PCT/US98/12681, filed on June 19, 1998, and entitled "Method and System for Incrementally Moving Teeth" ~~(attorney docket number 18563-000120)~~, which ~~claims priority from~~ is a continuation-in-part of U.S. patent application 08/947,080, filed on October 8, 1997, now U.S. Patent No. 5,975,893 which claims priority from U.S. provisional application 60/050,342, filed on June 20, 1997, all of which are incorporated by reference into this application.

TECHNOLOGICAL FIELD

The invention relates generally to the fields of dentistry and orthodontics and, more particularly, to subdividing a digital model of a patient's dentition.

BACKGROUND

Two-dimensional (2D) and three-dimensional (3D) digital image technology has recently been tapped as a tool to assist in dental and orthodontic treatment. Many treatment providers use some form of digital image technology to study the dentitions of patients. U.S. patent application 09/169,276, incorporated by reference above, describes the use of 2D and 3D image data in forming a digital model of a patient's dentition, including models of individual dentition components. Such models are useful, among other things, in developing an orthodontic treatment plan for the patient, as well as in creating one or more orthodontic appliances to implement the treatment plan.

SUMMARY

The inventors have developed several computer automated techniques for subdividing, or segmenting, a digital dentition model into models of individual dentition components. These dentition components include, but are not limited to, tooth crowns, tooth roots, and gingival regions. The segmentation techniques include both human assisted and fully automated techniques. Some of the human assisted techniques allow a human user to provide "algorithmic hints" by identifying certain features in the digital dentition model. The identified features then serve as a basis for automated segmentation. Some techniques act on a volumetric 3D image model, or "voxel representation," of the dentition, and other techniques act on a geometric 3D model, or "geometric representation."

In one aspect, a computer implementing the invention receives a data set that forms a three-dimensional (3D) representation of the patient's dentition, applies a test to the data set to identify data elements that represent portions of the individual component, and creates a digital model of the individual component based upon the identified data elements. Some implementations require the computer to identify data elements that form one or more 2D cross-sections of the dentition in one or more 2D planes intersecting the dentition. In many of these embodiments, these 2D planes are roughly parallel to the dentition's occlusal plane. The computer analyzes the features of the 2D cross-sections to identify data elements that correspond to the individual component to be modeled. For example, one technique requires the computer to identify cusps in the 2D cross-sectional surface of the dentition, where the cusps represent the locations of an interproximal margin between teeth in the dentition. One variation of this technique allows the computer to confine its search for cusps in one 2D plane to areas in the vicinity of cusps already identified on another 2D plane. Another variation allows the computer to link cusps on adjacent 2D planes to form a solid surface representing the interproximal margin. Some embodiments allow the computer to receive input from a human user identifying the cusp locations in one or more of the 2D cross sections.

Other embodiments require the computer to identify data elements that represent a structural core, or skeleton, of each individual component to be modeled. The computer creates the model by linking other data elements representing the individual component to the structural core.

In another aspect, a computer implementing the invention receives a three-dimensional (3D) data set representing the patient's dentition, applies a test to identify data elements that represent an interproximal margin between two teeth in the dentition, and applies another computer-implemented test to select data elements that lie on one side of the interproximal margin for inclusion in the digital model. Some implementations require the computer to identify data elements that form one or more 2D cross-sections of the dentition in one or more 2D planes intersecting the dentition roughly parallel to the dentition's occlusal plane.

In another aspect, a computer implementing the invention receives a 3D data set representing at least a portion of the patient's dentition, including at least a portion of a tooth and gum tissue surrounding the tooth; applies a test to identify data elements lying on a gingival boundary that occurs where the tooth and the gum tissue meet; and applies a test to the data elements lying on the boundary to identify other data elements representing portions of the tooth.

Other embodiments and advantages are apparent from the detailed description and the claims below.

DESCRIPTION OF THE DRAWINGS

FIGS. 1A, 1B, and 2 are partial views of a dentition model as displayed on a computer monitor and segmented with a human operated saw tool.

FIG. 3 is a partial view of a dentition model as displayed on a computer monitor and segmented with a human operated eraser tool.

FIG. 4 is a view of a dentition model for which a feature skeleton has been identified.

FIGS. 5 and 6 are flow diagram for a feature skeleton analysis technique used in segmenting a dentition model.

FIG. 7A is a horizontal 2D cross-sectional view of a dentition model.

FIG. 7B is a side view of a dentition model intersected by several 2D planes.

FIG. 8 is a flow diagram for a 2D slice analysis technique used in segmenting a dentition model.

FIGS. 9 and 10A through 10C each shows a group of voxels in a 2D slice of a dentition model.

FIG. 11 is a flow chart for an automatic cusp detection technique used in segmenting a dentition model.

FIG. 12 is a horizontal 2D cross-section of a dentition model illustrating a neighborhood filtered automatic cusp detection technique used in segmenting the dentition model.

FIG. 13 is shows two groups of voxels in a 2D slice of a dentition model illustrating the neighborhood filtered automatic cusp detection technique.

FIG. 14 is a flow chart for the neighborhood filtered automatic cusp detection technique.

FIG. 15 is a horizontal 2D cross-section of a dentition model illustrating an arch curve fitting technique used in segmenting the dentition model.

FIG. 16 is a flow chart for the arch curve fitting technique.

FIG. 17 is a horizontal 2D cross-section of a dentition model illustrating a curve creation technique for use with the arch curve fitting technique.

FIG. 18 is a flow diagram for the curve creation technique.

FIGS. 19A and 19B are a perspective view and a vertical 2D cross-sectional view of a dentition model illustrating another technique for use in segmenting the dentition model.

FIGS. 20 and 21 are flow diagrams of this technique.

FIG. 22 is a vertical 2D cross-sectional view of a dentition model illustrating a gingival margin detection technique for use in segmenting the dentition model.

FIG. 23 shows a group of voxels in a 2D slice of a dentition model illustrating a gingival margin detection technique.

FIG. 24 is a flow diagram for the gingival margin detection technique.

DETAILED DESCRIPTION

U.S. patent application 09/169,276 describes techniques for generating a 3D digital model of a patient's dentition, including the crowns and roots of the patients teeth as well as the surrounding gum tissue. One such technique involves creating a physical model of the dentition from a material such as plaster and then digitally imaging the model with a laser scanner or a destructive scanning system. The described techniques are used to produce a volumetric 3D image model ("volume element representation" or "voxel representation") and a geometric 3D surface model ("geometric model") of the dentition. The techniques described below act on one or both of these types of 3D dentition models. In creating a voxel representation, the physical model is usually embedded in a potting material that contrasts sharply with the color of the model to enhance detection of the dentition features. A white dentition model embedded in a black potting material provides the sharpest contrast. A wide variety of information is used to enhance the 3D model, including data taken from photographic images, 2D and 3D x-rays scans, computed tomography (CT) scans, and magnetic resonance imaging (MRI) scans of the patient's dentition.

Some computer-implemented techniques for segmenting a 3D dentition model into models of individual dentition components require a substantial amount of human interaction with the computer. One such technique, which is shown in FIGS. 1A, 1B, and 2, provides a graphical user interface with a feature that imitates a conventional saw, allowing the user to identify components to be cut away from the dentition model 100. The graphical user interface provides a rendered 3D image 100 of the dentition model, either at one or more static views from predetermined positions, as shown in FIGS. 1A and 1B, or in a "full 3D" mode that allows the user to alter the viewing angle, as shown in FIG. 2. The saw tool is implemented as a set of mathematical control points 102, represented graphically on the rendered image 100, which define a 3D cutting surface 104 that intersects the volumetric or geometric dentition model. The computer subdivides the data elements in the dentition model by performing a surface intersection operation between the 3D cutting surface 104 and the dentition model. The user sets the locations of the mathematical control points, and thus the geometry and position of the 3D cutting surface, by manipulating the control points in the graphical display with an input device, such as a mouse. The computer provides a visual representation 104 of the cutting surface on the display to assist the user in fitting the surface around the individual component to be separated. Once the intersection operation is complete, the computer creates a model of the individual component using the newly segmented data elements.

Another technique requiring substantial human interaction, shown in FIG. 3, is a graphical user interface with a tool that imitates a conventional eraser. The eraser tool allows the user to isolate an individual dentition component by removing portions of the dentition model that surround the individual component. The eraser tool is implemented as a 3D solid 110, typically having the shape of a rectangular prism, or a curved surface that matches the shape of a side surface of a tooth. The solid is made as small as possible, usually only a single voxel thick, to minimize degradation of the data set. As with the saw technique above, the graphical user interface presents the user with a rendered 3D image 112 of the dentition model at one or more predetermined static views or in a full 3D mode. The user identifies portions of the dentition model for removal by manipulating a graphical representation 110 of the 3D solid with an input device. In alternative embodiments, the computer either removes the identified portions of the dentition model as the user moves the eraser 112, or the computer waits until the user stops moving the eraser and provides an instruction to remove the identified portions. The computer updates the display in real time to show the path 114 of the eraser through the dentition model.

Other computer-implemented segmentation techniques require little or no human interaction during the segmentation process. One such technique, which is illustrated in FIG. 4, involves the application of conventional "feature skeleton" analysis to a volumetric representation of the dentition model. This technique is particularly useful in identifying and modeling individual teeth. In general, a computer applying this technique identifies a core of voxels, that forms a skeleton 122 for the dentition 120. The skeleton 122 roughly resembles the network of biological nerves within patient's teeth. The computer then divides the skeleton 122 into branches 124, each containing voxels that lie entirely within one tooth. One technique for identifying the branches is by defining a plane 126 that cuts through the skeleton 122 roughly parallel to the occlusal plane of the patient's dentition ("horizontal plane"). Each branch 124 intersects the horizontal plane 126 at one or more points, or clusters, that are relatively distant from the clusters associated with the other branches. The computer forms the individual tooth models by linking other voxels to the appropriate branches 124 of the skeleton.

FIG. 5 describes a particular technique for forming a skeleton in the dentition model. The computer first identifies the voxels in the dentition model that represent the tooth surfaces (step 130). For a voxel representation that is created from a physical model embedded in a sharply contrasting material, identifying the tooth surfaces is as simple as identifying the voxels at which sharp changes in image value occur, as described in U.S. patent application 09/169,276. The computer then calculates, for each voxel in the model, a distance measure indicating the physical distance between the voxel and the nearest tooth surface (step 132). The computer identifies the voxels with the largest distance measures and labels each of these voxels as forming a portion of the skeleton (step 134). Feature skeleton analysis techniques are described in more detail in the following publications: (1) Gagvani and Silver, "Parameter Controlled Skeletons for 3D Visualization," Proceedings of the IEEE Visualization Conference (1997); (2) Bertrand, "A Parallel Thinning Algorithm for Medial Surfaces," Pattern Recognition Letters, v. 16, pp. 979-986 (1995); (3) Mukherjee, Chatterji, and Das, "Thinning of 3-D Images Using the Safe Point Thinning Algorithm (SPTA)," Pattern Recognition Letters, v. 10, pp. 167-173 (1989); (4) Niblack, Gibbons, and Capson, "Generating Skeletons and Centerlines from the Distance Transform," CVGIP: Graphical Models and Image Processing, v. 54, n. 5, pp. 420-437 (1992).

Once a skeleton has been identified, the computer uses the skeleton to divide the dentition model into 3D models of the individual teeth. FIG. 6 shows one technique for doing so. The computer first identifies those portions of the skeleton that are associated with each individual tooth. To do so, the computer defines a plane that is roughly parallel to the dentition's occlusal surface and that intersects the skeleton near its base (step 136). The computer then identifies points at which the plane and the skeleton intersect by identifying each voxel that lies on both the skeleton and the plane (step 138). In general, a single tooth includes all of the voxels that lie in a particular branch of the skeleton; and because the plane intersects the skeleton near its base, voxels that lie together in a branch of the skeleton usually cluster together on the intersecting plane. The computer is able to locate the branches by identifying voxels on the skeleton that lie within a particular distance of each other on the intersecting plane (step 140). The computer then identifies and labels all voxels on the skeleton that belong to each branch (step 142).

Once the branches are identified, the computer links other voxels in the model to the branches. The computer begins by identifying a reference voxel in each branch of the skeleton (step 144). For each reference voxel, the computer selects an adjacent voxel that does not lie on the skeleton (step 146). The computer then processes the selected voxel, determining whether the voxel lies outside of the dentition, *i.e.*, whether the associated image value is above or below a particular threshold value (step 148); determining whether the voxel already is labeled as belonging to another tooth (step 150); and determining whether the voxel's distance measure is greater than the distance measure of the reference voxel (step 152). If none of these conditions is true, the computer labels the selected voxel as belonging to the same tooth as the reference voxel (step 154). The computer then repeats this test for all other voxels adjacent to the reference voxel (step 156). Upon testing all adjacent voxels, the computer selects one of the adjacent voxels as a new reference point, provided that the adjacent voxel is labeled as belonging to the same tooth, and then repeats the test above for each untested voxel that is adjacent to the new reference point. This process continues until all voxels in the dentition have been tested.

FIGS. 7A and 7B illustrate another technique for identifying and segmenting individual teeth in the dentition model. This technique, called "2D slice analysis," involves dividing the voxel representation of the dentition model into a series of parallel 2D planes 160, or slices, that are each one voxel thick and that are roughly parallel to the dentition's occlusal plane. Each of the 2D slices 160 includes a 2D cross-section 162 of the dentition, the surface 164 of which represents the lingual and buccal surfaces of the patient's teeth and/or gums. The computer inspects the cross-section 162 in each 2D slice 160 to identify voxels that approximate the locations of the interproximal margins 166 between the teeth. These voxels lie at the tips of cusps 165 in the 2D cross-sectional surface 164. The computer then uses the identified voxels to create 3D surfaces 168 intersecting the dentition model at these locations. The computer segments the dentition model along these intersecting surfaces 168 to create individual tooth models.

FIG. 8 describes a particular implementation of the 2D slice analysis technique. The computer begins by identifying the voxels that form each of the 2D slices (step 170). The computer then identifies, for each 2D slice, the voxels that represent the buccal and lingual surfaces of the patient's teeth and gums (step 172) and defines a curve that includes all of these voxels (step 174). This curve represents the surface 164 of the 2D cross-section 162.

The computer then calculates the rate of curvature (*i.e.*, the derivative of the radius of curvature) at each voxel on the 2D cross-sectional surface 164 (step 176) and identifies all of the voxels at which local maxima in the rate of curvature occur (step 178). Each voxel at which a local maximum occurs represents a "cusp" in the 2D cross-sectional surface 164 and roughly coincides with an interproximal margin between teeth. In each 2D slice, the computer identifies pairs of these cusp voxels that correspond to the same interproximal margin (step 180), and the computer labels each pair to identify the interproximal margin with which it is associated (step 182). The computer then identifies the voxel pairs on all of the 2D slices that represent the same interproximal margins (step 184). For each interproximal margin, the computer fits a 3D surface 168 approximating the geometry of the interproximal margin among the associated voxel pairs (step 186).

FIG. 9 illustrates one technique for creating the 3D surfaces that approximate the interproximal margins. For each pair of cusp voxels 190a-b in a 2D slice that are associated with a particular interproximal region, the computer creates a line segment 192 bounded by these cusp voxels 190a-b. The computer changes the colors of the voxels in the line segment, including the cusp voxels 190a-b that bound the segment, to contrast with the other voxels in the 2D slice. The computer creates line segments in this manner in each successive 2D slice, forming 3D surfaces that represent the interproximal regions. All of the voxels that lie between adjacent ones of these 3D surfaces represent an individual tooth.

FIGS. 10A through 10C illustrate a refinement of the technique shown in FIG. 9. The refined technique involves the projection of a line segment 200 from one slice onto a line segment 206 on the next successive slice to form, for the associated interproximal margin, a 2D area bounded by the cusp voxels 202a-b, 204a-b of the line segments 200, 206. If the line segments 200, 206 are oriented such that any voxel on one segment 200 is not adjacent to a voxel on the other segment 206, as shown in FIG. 10A, then the resulting 3D surface is discontinuous, leaving unwanted "islands" of white voxels 208, 210.

The computer eliminates these discontinuities by creating two new line segments 212, 214, each of which is bounded by one cusp voxel 202a-b, 204a-b from each original line segment 200, 206, as shown in FIG. 10B. The computer then eliminates the islands between the new line segments 212, 214 by changing the colors of all voxels between the new line segments 212, 214, as shown in FIG. 10C.

Automated segmentation is enhanced through a technique known as "seed cusp detection."

The term "seed cusp" refers to a location at which an interproximal margin meets the patient's gum tissue. In a volumetric representation of the patient's dentition, a seed cusp for a particular interproximal margin is found at the cusp voxel that lies closest to the gumline. By applying the seed cusp detection technique of the 2D slice analysis, the computer is able to identify all of the seed cusp voxels in the 3D model automatically.

FIG. 11 shows a particular implementation of the seed cusp detection technique, in which the computer detects the seed cusps by identifying each 2D slice in which the rate of curvature of a cusp first falls below a predetermined threshold value. The computer begins by selecting a 2D slice that intersects all of the teeth in the arch (step 220). The computer attempts to select a slice that is near the gingival regions but that does not include any voxels representing gingival tissue. The computer then identifies all of the cusp voxels in the 2D slice (step 222). If the rate of curvature of the 2D cross-section at any of the cusp voxels is less than a predetermined threshold value, the computer labels that voxel as a seed cusp (step 224). The computer then selects the next 2D slice, which is one voxel layer closer to the gingival region (step 226), and identifies all of the cusp voxels that are not associated with a cusp for which the computer has already identified a seed cusp (step 228). If the rate of curvature of the 2D cross-section is less than the predetermined threshold value at any of these cusp voxels, the computer labels the voxel as a seed cusp (step 230) and proceeds to the next 2D slice. The computer continues in this manner until a seed cusp voxel has been identified for each cusp associated with an interproximal margin (step 232).

FIGS. 12, 13, and 14 illustrate a technique, known as "neighborhood-filtered cusp detection," by which the computer focuses its search for cusps on one 2D slice to neighborhoods 244, 246 of voxels defined by a pair of previously detected cusp voxels 240, 242 on another 2D slice. Upon detecting a pair of cusp voxels 240, 242 in a 2D slice at level N (step 250), the computer defines one or more neighborhoods 244, 246 that include a predetermined number of voxels surrounding the pair (step 252). The computer then projects the neighborhoods onto the next 2D slice at level N+1 by identifying the voxels on the next slice that are immediately adjacent the voxels in the neighborhoods on the original slice (step 254). The neighborhoods are made large enough to ensure that they include the cusp voxels on the N+1 slice. In the example of FIG. 13, each cusp voxel 240, 242 lies at the center of a neighborhood 244, 246 of twenty-five voxels arranged in a 5 x 5 square.

In searching for the cusp voxels on the N+1 slice, the computer tests the image values for all voxels in the projected neighborhoods to identify those associated with the background image and those associated with the dentition (step 256). In the illustrated example, voxels in the background are black and voxels in the dentition are white. The computer identifies the cusp voxels on the N+1 slice by locating the pair of black voxels in the two neighborhoods that lie closest together (step 258). The computer then repeats this process for all remaining slices (step 259).

FIGS. 15 and 16 illustrate another technique, known as "arch curve fitting," for identifying interproximal margins between teeth in the dentition. The arch curve fitting technique, which also applies to 2D cross-sectional slices of the dentition, involves the creation of a curve 260 that fits among the voxels on the 2D cross-sectional surface 262 of the dentition arch 264. A series of closely-spaced line segments 268, each bounded by the cross-sectional surface 268, are formed along the curve 260, roughly perpendicular to the curve 260, throughout the 2D cross-section 264. In general, the shortest of these line segments 268 lie on or near the interproximal margins; thus computer identifies the cusps that define the interproximal margins by determining the relative lengths of the line segments 268.

When applying the arch curve fitting technique, the computer begins by selecting a 2D slice (step 270) and identifying the voxels associated with the surface 262 of the cross-sectional arch 264 (step 272). The computer then defines a curve 260 that fits among the voxels on the surface 262 of the arch (step 274). The computer creates the curve using any of a variety of techniques, a few of which are discussed below. The computer then creates a series of line segments that are roughly perpendicular to the curve and are bounded by the cross-sectional surface 262 (step 276). The line segments are approximately evenly spaced with a spacing distance that depends upon the required resolution and the acceptable computing time. Greater resolution leads to more line segments and thus greater computing time. In general, a spacing on the order of 0.4 mm is sufficient in the initial pass of the arch curve fitting technique.

The computer calculates the length of each line segment (step 278) and then identifies those line segments that form local minima in length (step 280). These line segments roughly approximate the locations of the interproximal boundaries, and the computer labels the voxels that bound these segments as cusp voxels (step 282). The computer repeats this process for each of the 2D slices (step 284) and then uses the cusp voxels to define 3D cutting surfaces that approximate the interproximal margins.

In some implementations, the computer refines the arch cusp determination by creating several additional sets of line segments, each centered around the arch cusps identified on the first pass. The line segments are spaced more narrowly on this pass to provide greater resolution in identifying the actual positions of the arch cusps.

The computer uses any of a variety of curve fitting techniques to create the curve through the arch. One technique involves the creation of a catenary curve with endpoints lying at the two ends 265, 267 (FIG. 15) of the arch. The catenary curve is defined by the equation $\mathbf{y} = \mathbf{a} + \mathbf{b} \cdot \cosh(\mathbf{cx})$, and the computer fits the curve to the arch by selecting appropriate values for the constants **a**, **b**, and **c**. Another technique involves the creation of two curves, one fitted among voxels lying on the front surface 271 of the arch, and the other fitted among voxels on the rear surface 273. A third curve, which guides the placement of the line segments above, passes through the middle of the area lying between the first two curves.

FIGS. 17 and 18 illustrate another technique for constructing a curve through the arch. This technique involves the creation of a series of initial line segments through the arch 264 and the subsequent formation of a curve 290 fitted among the midpoints of these line segments This curve 290 serves as the arch curve in the arch curve fitting technique described above.

In applying this technique, the computer first locates an end 265 of the arch (step 300) and creates a line segment 291 that passes through the arch 264 near this end 265 (step 301). The line segment 291 is bounded by voxels 292a-b lying on the surface of the arch. The computer then determines the midpoint 293 of the line segment 291 (step 302), selects a voxel 294 located particular distance from the midpoint 293 (step 304), and creates a second line segment 295 that is parallel to the initial line segment 291 and that includes the selected voxel 294 (step 306). The computer then calculates the midpoint 296 of the second segment 295 (step 308) and rotates the second segment 295 to the orientation 295' that gives the segment its minimum possible length (step 309). In some cases, the computer limits the second segment 295 to a predetermined amount of rotation (*e.g.*, ± 10°).

The computer then selects a voxel 297 located a particular distance from the midpoint 296 of the second segment 295 (step 310) and creates a third line segment 298 that is parallel to the second line segment 295 and that includes the selected voxel 297 (step 312). The computer calculates the midpoint 299 of the third segment 298 (step 314) and rotates the segment 298 to the orientation 298' that gives the segment its shortest possible length (step 316). The computer continues adding line segments in this manner until the other end of the cross-sectional arch is reached (step 318). The computer then creates a curve that fits among the midpoints of the line segments (step 320) and uses this curve in applying the arch fitting technique described above.

FIGS. 19A, 19B and 20 illustrate an alternative technique for creating 3D surfaces that approximate the geometries and locations of the interproximal margins in the patient's dentition. This technique involves the creation of 2D planes that intersect the 3D dentition model at locations that approximate the interproximal margins. In general, the computer defines a series of planes, beginning with an initial plane 330 at one end 331 of the arch 332, that are roughly perpendicular to the occlusal plane of the dentition model ("vertical" planes). Each plane intersects the dentition model to form a 2D cross-section 334. If the planes are spaced sufficiently close to each other, the planes with the smallest cross-sectional areas approximate the locations of the interproximal margins in the dentition. The computer locates the interproximal regions more precisely by rotating each plane about two orthogonal axes 336, 338 until the plane reaches the orientation that yields the smallest possible cross-sectional area.

In one implementation of this technique, the computer first identifies one end of the arch in the dentition model (step 340). The computer then creates a vertical plane 330 through the arch near this end (step 342) and identifies the centerpoint 331 of the plane 330 (step 344). The computer then selects a voxel located a predetermined distance from the centerpoint (step 345) and creates a second plane 333 that is parallel to the initial plane and that includes the selected voxel (step 346). The computer calculates the midpoint of the second plane (step 348) and rotates the second plane about two orthogonal axes that intersect at the midpoint (step 350). The computer stops rotating the plane upon finding the orientation that yields the minimum cross-sectional area. In some cases, the computer limits the plane to a predetermined amount of rotation (*e.g.*, ± 10° about each axis). The computer then selects a voxel located a particular distance from the midpoint of the second plane (step 352) and creates a third plane that is parallel to the second plane and that includes the selected voxel (step 354). The computer calculates the midpoint of the third plane (step 356) and rotates the plane to the orientation that yields the smallest possible cross-sectional area (step 357). The computer continues adding and rotating planes in this manner until the other end of the arch is reached (step 358). The computer identifies the planes at which local minima in cross-sectional area occur and labels these planes as "interproximal planes," which approximate the locations of the interproximal margins (step 360).

One variation of this technique, described in FIG. 21, allows the computer to refine its identification of interproximal planes by creating additional, more closely positioned planes in areas around the planes labeled as interproximal. The computer first creates a curve that fits among the midpoints of the planes labeled as interproximal planes (step 372) and then creates a set of additional planes along this curve (step 374). The additional planes are not evenly spaced along the curve, but rather are concentrated around the interproximal margins. The planes in each interproximal area are spaced very closely (*e.g.*, 0.05 mm from each other). The computer rotates each of the newly constructed planes about two orthogonal axes until the plane reaches its minimum cross-sectional area (step 376). The computer then selects the plane in each cluster with the smallest cross-sectional area as the plane that most closely approximates the interproximal margin (step 378).

FIGS. 22, 23, and 24 illustrate a technique for identifying the gingival margin that defines the boundary between tooth and gum in the patient's dentition. This technique involves the creation of a series of horizontal 2D planes 380, or slices, that intersect the dentition model roughly perpendicular to the occlusal plane (see FIG. 19A). The cross-sectional surface 382 of the dentition model in each of these planes 380 includes cusps 384, 386 that represent the gingival margin. The computer identifies the gingival margin by applying one or more of the cusp detection techniques described above.

One technique is very similar to the neighborhood filtered cusp detection technique described above, in that voxel neighborhoods 388, 390 are defined on one of the 2D planes to focus the computer's search for cusps on an adjacent 2D plane. Upon detecting a pair of cusps 384, 386 on one 2D plane (step 400), the computer defines one or more neighborhoods 388, 390 to include a predetermined number of voxels surrounding the pair (step 402). The computer projects the neighborhoods onto an adjacent 2D plane by identifying the voxels on the adjacent plane that correspond to the voxels in the neighborhoods 388, 390 on the original plane (step 404). The computer then identifies the pair of black voxels that lie closest together in the two neighborhoods on the adjacent plane, labeling these voxels as lying in the cusp (step 406). The computer repeats this process for all remaining planes (step 408).

Many of these automated segmentation techniques are even more useful and efficient when used in conjunction with human-assisted techniques. For example, techniques that rely on the identification of the interproximal or gingival margins function more quickly and effectively when a human user first highlights the interproximal or gingival cusps in a graphical representation of the dentition model. One technique for receiving this type of information from the user is by displaying a 2D or 3D representation and allowing the user to highlight individual voxels in the display. Another technique allows the user to scroll through a series of 2D cross-sectional slices, identifying those voxels that represent key features such as interproximal or gingival cusps. Some of these techniques rely on user interface tools such as cursors and bounding-box markers.

In many instances, the computer creates proposals for segmenting the dentition model and then allows the user to select the best alternative. For example, one version of the arch curve fitting technique requires the computer to create a candidate catenary or spline curve, which the user is allowed to modify by manipulating the mathematical control parameters. Other techniques involve displaying several surfaces that are candidate cutting surfaces and allowing the user to select the appropriate surfaces.

Some implementations of the invention are realized in digital electronic circuitry, such as an application specific integrated circuit (ASIC); others are realized in computer hardware, firmware, and software, or in combinations of digital circuitry and computer components. The invention is usually embodied, at least in part, as a computer program tangibly stored in a machine-readable storage device for execution by a computer processor. In these situations, methods embodying the invention are performed when the processor executes instructions organized into program modules, operating on input data and generating output. Suitable processors include general and special purpose microprocessors, which generally receive instructions and data from read-only memory and/or random access memory devices. Storage devices that are suitable for tangibly embodying computer program instructions include all forms of non-volatile memory, including semiconductor memory devices, such as EPROM, EEPROM, and flash memory devices; magnetic disks such as internal hard disks and removable disks; magneto-optical disks; and CD-ROM.

The invention has been described in terms of particular embodiments. Other embodiments are within the scope of the following claims.